802.3bp Sleep/Wake-up Specification

TC10 - OPEN Sleep/Wake-up Specification for Automotive Gigabit Ethernet

Author & Company	Philip Axer (NXP), Dachin Tseng (Realtek), Wei Lou (Broadcom), Cliff Fung (Marvell)
Title	802.3bp Sleep/Wake-up Specification
Version	1.0
Date	27.10.2021
Status	Final
Restriction Level	Public Release

Version Control of Document

Version	Author	Description	Date
0.1	Dachin Tseng	First Draft Version	2019/8/12
0.2	Dachin Tseng	Add 1. LPS and WUR encode/decode in OAM 2. WUP to define tx_mode, symbol rate and transmitting duration 3. WUP amplitude range is TBD	2019/11/12
0.3	Philip Axer	Added 1. Wakeup electrical interface recommendations Updated 1. PHY Control 2. Link Sync 3. Autoneg	2020/08/19
0.4	Philip Axer	1. Removed high level power sequencing FSM from Introduction 2. Fixes in PHY Control FSM 3. Fixes in Link Sync FSM 4. Fixed in AN FSM 5. Changed recommendation of electrical wakeup interface 6. Added statement on pulse width of wake outputs 7. Added ForceSleep service primitive 8. Added paragraph on WUP compatibility with 100Base-T1 9. Other minor fixes	2020/10/21
0.5	Philip Axer	1. Updated timing requirements to match ISO definitions 2. Updates wakeup/sleep electrical interface paragraph to match the table 3. Added information which scrambler type is used for WUP 4. Added optional WUP transmission on startups 5. Removed WUP compatibility table and restricted WUP to a specific speed 6. Added details to lps_timer_done definition 7. Revised power state diagram with added SLEEP_WAIT state	2021/02/03
0.6	Philip Axer	1. Added missing definition of sleep_wait_timer 2. Aggregated times for T_Initialization and T_Powersupply into a joint 15 ms interval 3. Made WakeupForward.indication and WakeupForward.request optional	2021/03/10

		4. Added resolution statement to WUP "autonegotiation result" in case there autoneg has not been performed 5. Changed link_sync_wd_timer and an_watchdog_timer to 40 ms 6. Added missing transition from SLEEP_SILENT to SEND_DATA 7. Changed definition of tx_lps_done 8. Added lps_timer(done) definition 9. Revised WUP amplitude to contain a lower bound 10. Revised Wakeup.indicate condition 11. Changed the TWU_WakeIO to not mention the now optional WakeupForward primitives	
0.7		1. Added timer tolerances 2. Changed wake voltage source from VDDIO to "VDDIO or VDD_AO" 3. Rephrased WUP pattern	2021/04/14
0.8		1. Updated editorial issues and typos 2. Updated PHY Control, LinkSync and AN state diagrams 3. Added tc10_link_start signal	2021/06/23
0.9		1. Figure 3: Changed start sendz_timer to sendz_minwait_timer and sendz_maxwait_timer 2. 9.2.1: Fixed typo link_sync_timer →link_sync_wd_timer 3. PHY Control: removed minwait_timer_done from SEND_DATA to SLEEP_SILENT transition 4. PHY Control: Added !tc10_start_link as gating criteria to transitions to DISABLE_TRANSMITTER and WAKE_TRANSMIT 5. 9.2.1: Fixed typo AN_send_wake_req → AN_send_wup_req	2021/07/02
0.10		1. Changed TC10 statemachine back to sleep<=FALSE during SLEEP_WAIT state 2. Clarify LPS command encoding to stop when tx_lps_done is asserted by adding &!tx_lps_done to the definition 3. Change TC10 state machine NORMAL state to include tx_lps_done = false	2021/07/28
1.0	Cliff Fung	1. Change status to public release	2021/10/27

Restriction level history of Document

Version	Restriction Level	Description	Date
0.1	OPEN Technical Members Only	Technical Members	2019/8/12
0.2	OPEN Technical Members Only	Technical Members	2019/11/12
0.3	OPEN Technical Members Only	Technical Members	2020/08/19
0.4	OPEN Technical Members Only	Technical Members	2020/10/21
0.5	OPEN Technical Members Only	Technical Members	2021/02/03
0.6	OPEN Technical Members Only	Technical Members	2021/03/10
0.7	OPEN Technical Members Only	Technical Members	2021/04/14
0.8	OPEN Technical Members Only	Technical Members	2021/06/23
0.9	OPEN Technical Members Only	Technical Members	2021/07/02
0.10	OPEN Technical Members Only	Technical Members	2021/07/28
1.0	Public Release	Public Release	2021/10/27

Contents

Foreword (Disclaimer)

OPEN Alliance Specification Copyright Notice and Disclaimer

A. OPEN Specification Ownership and Usage Rights

As between OPEN Alliance and OPEN Alliance Members whose contributions were incorporated in this OPEN Specification (the "Contributing Members"), the Contributing Members own the worldwide copyrights in and to their given contributions. Other than the Contributing Members' contributions, OPEN Alliance owns the worldwide copyrights in and to compilation of those contributions forming this OPEN Specification. For OPEN Alliance Members (as that term is defined in the OPEN Alliance Bylaws), OPEN Alliance permits the use of this OPEN Specification on the terms in the OPEN Alliance Intellectual Property Rights Policy and the additional applicable terms below. For non-members of OPEN Alliance, OPEN Alliance permits the use of this OPEN Specification on the terms in the OPEN Alliance Specification License Agreement (available here: http://www.opensig.org/Automotive-Ethernet-Specifications/) and the additional applicable terms below. The usage permissions referenced and described here relate only to this OPEN Specification and do not include or cover a right to use any specification published elsewhere and referred to in this OPEN Specification. By using this OPEN Specification, you hereby agree to the following terms and usage restrictions:

A.i. Rights and Usage Restrictions Specific to OPEN Alliance Members

FOR OPEN ALLIANCE MEMBERS ONLY: In addition to the usage terms and restrictions granted to Members in the OPEN Alliance Intellectual Property Rights Policy, Members' use of this OPEN Specification is subject this Copyright Notice and Disclaimer. Members of OPEN Alliance have the right to use this OPEN Specification solely (i) during the term of a Member's membership in OPEN Alliance and subject to the Member's continued membership in good standing in OPEN Alliance; (ii) subject to the Member's continued compliance with the OPEN Alliance governance documents, Intellectual Property Rights Policy, and the applicable OPEN Alliance Promoter or Adopter Agreement, as applicable; and (iii) for internal business purposes and solely to use the OPEN Specification for implementation of this OPEN Specification in the Member's products and services, but only so long as Member does not distribute, publish, display, or transfer this OPEN Specification to any third party, except as expressly set forth in Section 11 of the OPEN Alliance Intellectual Property Rights Policy. Except and only to the extent required to use this OPEN Specification internally for implementation of this OPEN Specification in Member's products and services, Member shall not modify, alter, combine, delete portions of, prepare derivative works of, or create derivative works based upon this OPEN Specification. If Member creates any modifications, alterations, or other derivative works of this OPEN Specification as permitted to use the same internally for implementation of this OPEN Specification in Member's products and services, all such modifications, alterations, or other derivative works shall be deemed part of, and licensed to such Member under the same restrictions as, this OPEN Specification. For the avoidance of doubt, Member shall not include all or any portion of this OPEN Specification in any other technical specification or technical material, product manual, marketing material, or any other material without OPEN Alliance's prior written consent. All rights not expressly granted to Member in the OPEN Alliance Intellectual Property Rights Policy are reserved;

A.ii. Rights and Usage Restrictions Specific to Non-members of OPEN Alliance

FOR NON-MEMBERS OF OPEN ALLIANCE ONLY: Use of this OPEN Specification by anyone who is not a Member in good standing of OPEN Alliance is subject to your agreement to the OPEN Alliance Specification License Agreement (available here: http://www.opensig.org/Automotive-Ethernet-Specifications/) and the additional terms in this Copyright Notice and Disclaimer. Non-members have the right to use this OPEN Specification solely (i) subject to the non-member's continued compliance with the OPEN Alliance Specification License Agreement; and (ii) for internal business purposes and solely to use the OPEN Specification for implementation of this OPEN Specification in the non-member's products and services, but only so long as non-member does not distribute, publish, display, or transfer this OPEN Specification to any third party, unless and only to the extent expressly set forth in the OPEN Alliance Specification License Agreement. Except and only to the extent required to use this OPEN Specification internally for implementation of this OPEN Specification in non-member's products and services, non-member shall not modify, alter, combine, delete portions of, prepare derivative works of, or create derivative works based upon this OPEN Specification. If non-member creates any modifications, alterations, or other derivative works of this OPEN Specification as permitted to use the same internally for implementation of this OPEN Specification in non-member's products and services, all such modifications, alterations, or other derivative works shall be deemed part of, and licensed to such non-member under the same restrictions as, this OPEN Specification. For the avoidance of doubt, non-member shall not include all or any portion of this OPEN Specification in any other technical specification or technical material, product manual, marketing material, or any other material without OPEN Alliance's prior written consent. All rights not expressly granted to non-member in the OPEN Alliance Specification License Agreement are reserved.

B. Terms Applicable to both Members and Non-members of OPEN Alliance

B.i. Patents, Trademarks, and other Rights:

OPEN Alliance has received no Patent Disclosure and Licensing Statements related to this OPEN Specification. Therefore, this OPEN Specification contains no specific disclaimer related to third parties that may require a patent license for their Essential Claims. Having said that, the receipt of this OPEN Specification shall not operate as an assignment of or license under any patent, industrial design, trademark, or other rights as may subsist in or be contained in or reproduced in this OPEN Specification; and the implementation of this OPEN Specification could require such a patent license from a third party. You may not use any OPEN Alliance trademarks or logos without OPEN Alliance's prior written consent.

B.ii. Disclaimers and Limitations of Liability:

THIS OPEN SPECIFICATION IS PROVIDED ON AN "AS IS" BASIS, AND ALL REPRESENTATIONS, WARRANTIES, AND GUARANTEES, EITHER EXPLICIT, IMPLIED, STATUTORY, OR OTHERWISE, ARE EXCLUDED AND DISCLAIMED UNLESS (AND THEN ONLY TO THE EXTENT THEY ARE) MANDATORY UNDER LAW. ACCORDINGLY, OPEN ALLIANCE AND THE CONTRIBUTING MEMBERS MAKE NO REPRESENTATIONS OR WARRANTIES OR GUARANTEES WITH REGARD TO THIS OPEN SPECIFICATION OR THE INFORMATION (INCLUDING ANY SOFTWARE) CONTAINED HEREIN. OPEN ALLIANCE AND ALL CONTRIBUTING MEMBERS HEREBY EXPRESSLY DISCLAIM ANY AND ALL SUCH EXPRESS, IMPLIED, STATUTORY, AND ALL OTHER REPRESENTATIONS,

WARRANTIES, AND GUARANTEES, INCLUDING WITHOUT LIMITATION ANY AND ALL WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR USE, TITLE, NON-INFRINGEMENT OF OR ABSENCE OF THIRD PARTY RIGHTS, AND/OR VALIDITY OF RIGHTS IN THIS OPEN SPECIFICATION; AND OPEN ALLIANCE AND THE CONTRIBUTING MEMBERS MAKE NO REPRESENTATIONS AS TO THE ACCURACY OR COMPLETENESS OF THIS OPEN SPECIFICATION OR ANY INFORMATION CONTAINED HEREIN. WITHOUT LIMITING THE FOREGOING, OPEN ALLIANCE AND/OR CONTRIBUTING MEMBERS HAS(VE) NO OBLIGATION WHATSOEVER TO INDEMNIFY OR DEFEND YOU AGAINST CLAIMS RELATED TO INFRINGEMENT OR MISAPPROPRIATION OF INTELLECTUAL PROPERTY RIGHTS.

OPEN ALLIANCE AND CONTRIBUTING MEMBERS ARE NOT, AND SHALL NOT BE, LIABLE FOR ANY LOSSES, COSTS, EXPENSES, OR DAMAGES OF ANY KIND WHATSOEVER (INCLUDING WITHOUT LIMITATION DIRECT, INDIRECT, SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE, AND/OR EXEMPLARY DAMAGES) ARISING IN ANY WAY OUT OF USE OR RELIANCE UPON THIS OPEN SPECIFICATION OR ANY INFORMATION HEREIN. NOTHING IN THIS DOCUMENT OPERATES TO LIMIT OR EXCLUDE ANY LIABILITY FOR FRAUD OR ANY OTHER LIABILITY WHICH IS NOT PERMITTED TO BE EXCLUDED OR LIMITED BY OPERATION OF LAW.

B.iii. <u>Compliance with Laws and Regulations</u>:

NOTHING IN THIS DOCUMENT OBLIGATES OPEN ALLIANCE OR CONTRIBUTING MEMBERS TO PROVIDE YOU WITH SUPPORT FOR, OR RELATED TO, THIS OPEN SPECIFICATION OR ANY IMPLEMENTED PRODUCTS OR SERVICES. NOTHING IN THIS OPEN SPECIFICATION CREATES ANY WARRANTIES OR GUARANTEES, EITHER EXPRESS OR IMPLIED, STATUTORY OR OTHERWISE, REGARDING ANY LAW OR REGULATION. OPEN ALLIANCE AND CONTRIBUTING MEMBERS EXPRESSLY DISCLAIM ALL LIABILITY, INCLUDING WITHOUT LIMITATION, LIABILITY FOR NONCOMPLIANCE WITH LAWS, RELATING TO USE OF THE OPEN SPECIFICATION OR INFORMATION CONTAINED HEREIN. YOU ARE SOLELY RESPONSIBLE FOR THE COMPLIANCE OF IMPLEMENTED PRODUCTS AND SERVICES WITH ANY SUCH LAWS AND REGULATIONS, AND FOR OBTAINING ANY AND ALL REQUIRED AUTHORIZATIONS, PERMITS, AND/OR LICENSES FOR IMPLEMENTED PRODUCTS AND SERVICES RELATED TO SUCH LAWS AND REGULATIONS WITHIN THE APPLICABLE JURISDICTIONS.

IF YOU INTEND TO USE THIS OPEN SPECIFICATION, YOU SHOULD CONSULT ALL APPLICABLE LAWS AND REGULATIONS. COMPLIANCE WITH THE PROVISIONS OF THIS OPEN SPECIFICATION DOES NOT CONSTITUTE COMPLIANCE TO ANY APPLICABLE LEGAL OR REGULATORY REQUIREMENTS. IMPLEMENTERS OF THIS OPEN SPECIFICATION ARE SOLELY RESPONSIBLE FOR OBSERVING AND COMPLYING WITH THE APPLICABLE LEGAL AND REGULATORY REQUIREMENTS. WITHOUT LIMITING THE FOREGOING, YOU SHALL NOT USE, RELEASE, TRANSFER, IMPORT, EXPORT, AND/OR RE-EXPORT THIS OPEN SPECIFICATION OR ANY INFORMATION CONTAINED HEREIN IN ANY MANNER PROHIBITED UNDER ANY APPLICABLE LAWS AND/OR REGULATIONS, INCLUDING WITHOUT LIMITATION U.S. EXPORT CONTROL LAWS.

B.iv. <u>Automotive Applications Only</u>: Without limiting the foregoing disclaimers or limitations of liability in any way, this OPEN Specification was developed for automotive applications only. This OPEN Specification has neither been developed, nor tested for, non-automotive applications.

B.v. <u>Right to Withdraw or Modify</u>:
OPEN Alliance reserves the right to (but is not obligated to) withdraw, modify, or replace this OPEN Specification at any time, without notice.

Introduction

This specification defines new service primitives provided by the ISO/OSI layer 1 (PHY) and supporting a controlled link shutdown and a fast global wake-up within an Ethernet network. Higher layers like the network management can access those service primitives to realize partial networking, where selected parts of a network are inactive. The coordination of switching off selected nodes of a network is handled by the network management and is not part of this specification. This partial networking concept relying on selective link shutdown and fast global wake-up is especially suited for automotive Ethernet networks.

The 802.3-2018: 1000Base-T1 specification does not define mechanisms for a controlled link shut-down and wake-up. Therefore the new service primitives defined in this specification can be regarded as a supplement to the 802.3-2018: 1000Base-T1 specification.

The new services primitives make use of LPS, WUR and WUP commands. When not using the new service primitives and commands, implementation of these extensions will not impact the interoperability to a “basic” 802.3-2018: 1000Base-T1 PHY.

1 Scope

The following are the objectives of the Sleep/Wake-up specification:

a) Comply with the CSMA/CD MAC

b) Comply with the specifications for the xMII (MII, RMII, RGMII etc.)

c) Support global network wake-up (incl. link start-up time) within less 250ms

d) Support wake-up process completely covered in ISO/OSI layer 1

e) Support controlled link shutdown to deactivate selective parts of network

f) Comply with Autosar network management

g) No unwanted wakeup in presence of interference noise

h) Applicable for 1000BASE-T1

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

Amendment 4: Physical Layer Specification and Management Parameters for 1Gb/s Operation over a Single Twisted-Pair Copper Cable", IEEE Std 802.3-2018

3 Terms and Definitions

For the purposes of this document, the following terms and definitions apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at http://www.electropedia.org/

4 Wakeup/Sleep electrical interface

PHYs supporting wakeup and sleep signalling over dedicated I/O pins should follow the following guidelines. Sleep/wakeup commands are not part of the xMII interface.

The following list shows examples of pins typically associated with wakeup and sleep functionality. Depending on the type of device and functionality not all pins are relevant.

Table 1 Example of wakeup related device pins

Pin name	Direction[1]	Function	Voltage Source	Mandatory
INH	OUT (Wired-OR)	Prevent external regulator from shutdown	VDD_AO	Recommended
LOCAL_WAKE	IN	Local wake input	VDDIO[2] or VDD_AO	Yes
WAKE_FWRD	OUT	Wake	VDDIO or VDD_AO	No
WAKE_IN_OUT	IN/OUT (Wired-OR)	Multiplex interface to support wake input and output over the same pin	VDDIO or VDD_AO	No[3]
VDD_AO	IN	Always-on supply, available during sleep to power wakeup detection functionality	VBAT or other available standby supply	Yes

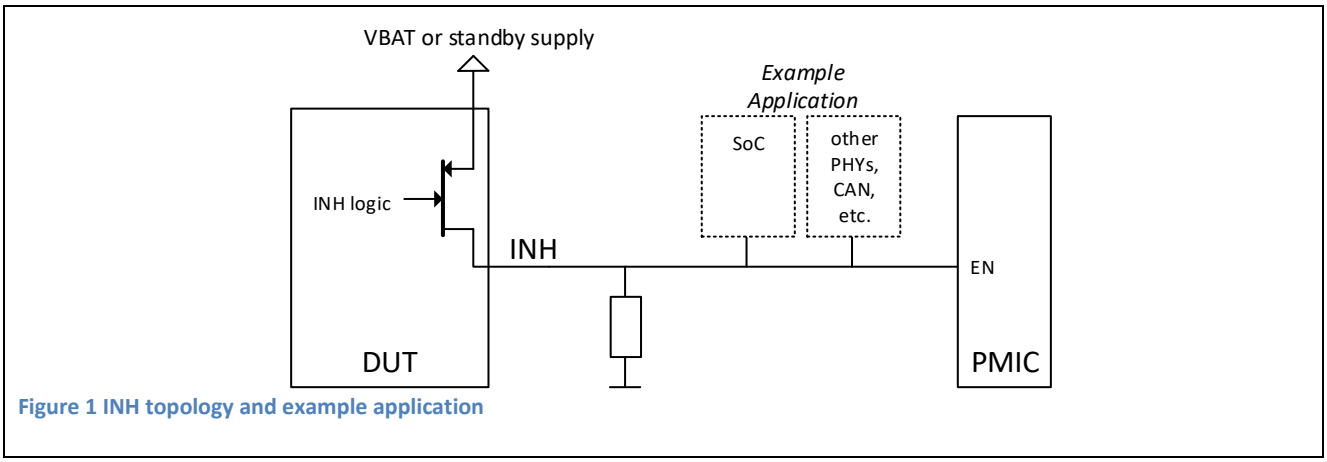


Figure 1 INH topology and example application

The interface shall support a local-wakeup input and may support a wakeup-forwarding output. For multi PHY designs these pins can be joined. Inhibit pins should be a high-side switch, which is pulling INH high during normal operation and high-Z during sleep. A dedicated wakeup-forwarding pin must be active-high.

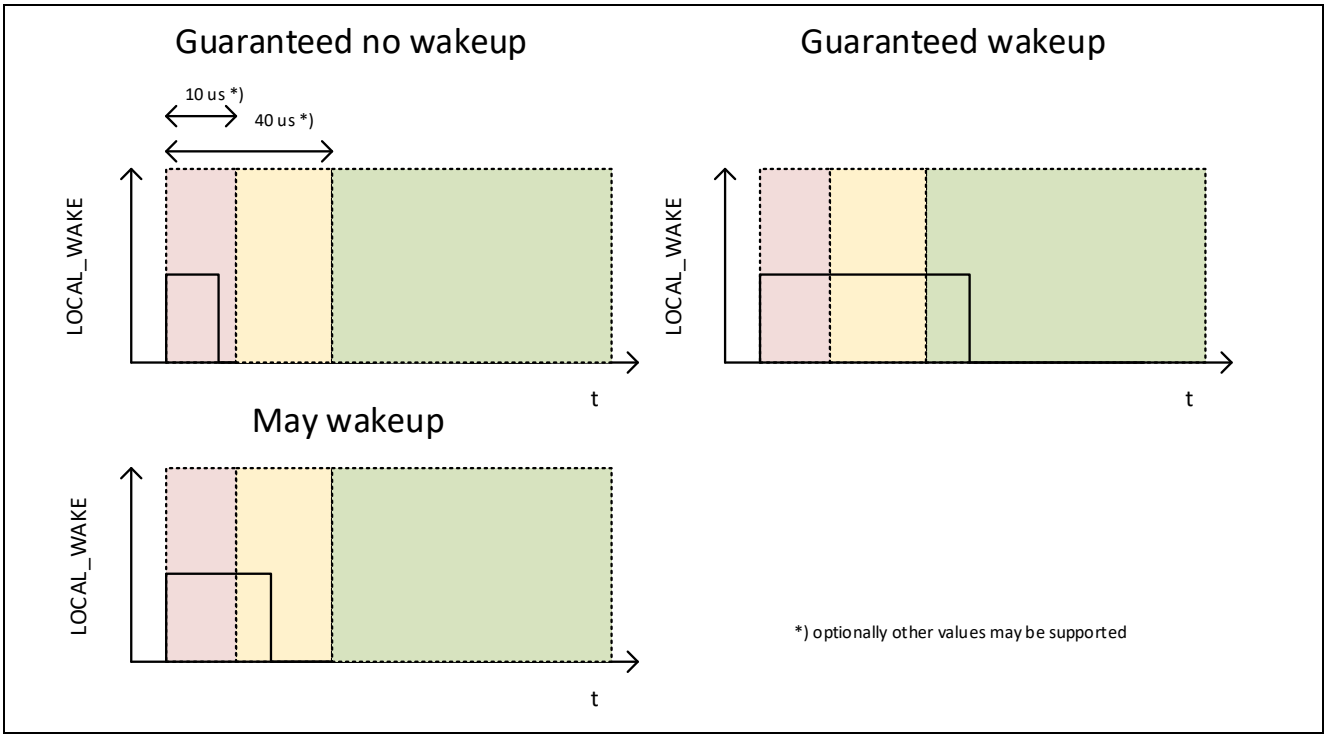
The I/O voltage is left to the implementer. Note that INH is supplied by a standby supply which is available during sleep.

In some automotive use-cases a glitching wakeup source is connected to the LOCAL_WAKE pin. In order to support such applications it is required to implement a detection threshold. Pulses with a duration of less than 10 us must not be detected as wakeup event and are to be ignored. A pulse duration of more than 40 us must be guaranteed to be detected and cause a wakeup. Note that pulses in this interval are undefined.

[1] From the perspective of the DUT
[2] Standard JEDEC voltage level recommended, no specific voltage level given in this spec
[3] WAKE_IN_OUT can optionally be used to replace dedicated LOCAL_WAKE and WAKE_FWRD and merge the functionality into a single pin.

From this follows that a local wakeup output pulse (originating from WAKE_IN_OUT, WAKE_FWRD or another source) must have a duration of at least 40 us to be reliably detected.

In case LOCAL_WAKE is fed through the wiring harness (support for slow legacy wake-up line) it is recommended to have the option to increase the rejection window (minimum of 10 ms).



5 Power consumption

The following guidelines on the device power consumption in sleep mode target typical Ethernet products such as single and multiport PHYs and switches.

A single-port PHY product should have a quiescence current of 35µA. A multi-port PHY or switch product should have a quiescence current of 25µA plus 10µA for each port[4].

6 Timing Behavior

The sleep and wake up process in a PHY shall fulfill the following requirements[5]:

6.1 ACK_timer

The time duration in SLEEP_ACK state shall expire in 8ms.

ACK_timer = 8 ms.

6.2 REQ_timer

The time duration in SLEEP_REQUEST state and SLEEP_SILENT state shall expire in 16ms.

REQ_timer = 16 ms.

6.3 TWU_Link_passive

The wake-up transmission time over a passive link (WUP) shall be less than 2 ms.

TWU_Link_passive < 2 ms.

6.4 TWU_Link_active

Wake-up transmission time over an active link (WUR) shall be less than 2 ms.

TWU_Link_active < 2 ms

6.5 TWU_Forwarding

For multiport devices it is possible to forward a wakeup from one physical port to another physical port. The Wake-up forwarding time is the time from receiving a wakeup WakeupForward.Indication on one physical port until a WakeupFoward.Request is generated on another physical port.

Wake-up forwarding time shall be less than 1 ms.

TWU_Forwarding < 1 ms

For a passive device, the time from Wakeup.request on the initiating device to Wakeup.Indication on the responding device is:

[4] More complex SoC products with other wakeup-capable interfaces may exceed these numbers, while still meeting this specification.

[5] For the mentioned timer values a 10 % tolerance is expected.

TWU_Link_Passive + T_Powersupply_stable + T_Initalization,

Where T_Powersupply_stable is the time from which the passive device requests power until the power supply is stable.

T_Initialization the time from when the power supply's stable voltage is reached until Wakeup.Indication is generated.

The sum of T_Powersupply_stable and T_Initalization shall be less than 15ms

6.6 TWU_WakeIO

The time TWU_WakeIO is defined from the generation of a Wakeup.request in one device to the reception of the corresponding Wakeup.indication in the other device when both devices are connected by using the electrical wakeup interface pins (for instance WAKE_FWRD or WAKE_IN_OUT).

TWU_WakeIO < 1 ms

7 Service Primitives and Interfaces

Beside the service primitives and interfaces, specified in IEEE802.3, new service primitives are provided by the 802.3 physical layer to the upper management layer. These services are needed to realize the sleep and wake-up behaviour.

The wakeup and sleep control information is transferred between the PMA, PCS and SMI and physical device pins. This document does not specify an SMI address layout.

7.1 Sleep.request

The purpose of the *Sleep.request* service primitive is to shut down a link in a controlled manner, without generating unwanted link failure interrupts.

7.2 SleepForce.request

The purpose of SleepForce is to force the device into Sleep state bypassing the sleep handshake.

7.3 Sleep.indication

The purpose of the Sleep.indication service primitive is to indicate a received sleep request.

7.4 SleepFail.indication

The purpose of the optional SleepFail.indication service primitive is to indicate an aborted or unsuccessful sleep handshake.

7.5 SleepAbort.request

The purpose of the *SleepAbort.request* service primitive is abort a received sleep request.

7.6 Wakeup.request

The purpose of the Wakeup.request service primitive is to generate a WUP or WUR command leading to a global wake-up within the Ethernet network.

7.7 Wakeup.indication

The purpose of the *Wakeup.indication* service primitive is to indicate a detected wake-up event. This includes a wakeup over a passive link, a wakeup over an active link as well as over a local wakeup pin. A *Wakeup.indication* can originate from MDI side (as WUP or WUR), from SMI side (over wakeup register) or over a physical pin (WAKE_IN, WAKE_IN_OUT).

7.8 Inhibit.indication

Signals the state of an optional power supply inhibit interface.

7.9 WakeupForward.indication

(optional)

This service primitive signals that a Wakeup forwarding event has been received over wake I/O functionality or MDI.

7.10 WakeupForward.request

(optional)

This service primitive signals that a wakeup event has been forwarded to this port as a consequence of a WakeupForward.indication on another port or through the wake I/O functionality.

8 Command Definitions

This specification defines three commands which are used to request a power down and signal a wakeup over an active as well as a passive link for 1000BASE-T1.

8.1 Low Power Sleep (LPS) and Wake-Up Request (WUR)

The Low Power Sleep (LPS) is a command to indicate a sleep request to the link partner. The LPS command is sent by a node requesting a transition to SLEEP, while the link is up.

The Wake-Up Request (WUR) is a command to indicate a wake-up request to the link partner. It can be sent by a node PHY or switch PHY to distribute the wake-up request over a link, which is already active.

The LPS and WUR are encode in 1000BASE-T1 PCS level Operations, Administration, and Maintenance (OAM). The OAM Frame is shown as follow:

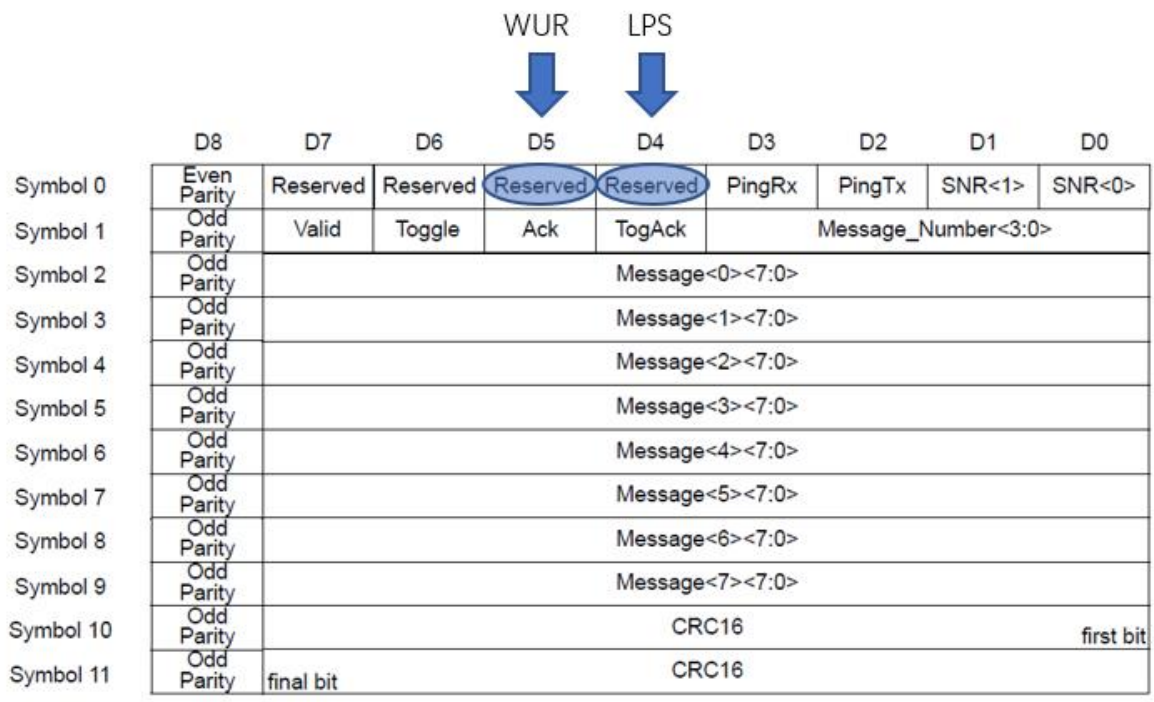


Figure 2 LPS and WUR coding in OAM frame

The bits LPS and WUR are defined as follows:

$$LPS = (\text{tx\_lps} = \text{TRUE}) \ \& \ (\text{loc\_wake\_req} = \text{FALSE}) \ \& \ (\text{link\_status} = \text{OK}) \ \& \ !\text{tx\_lps\_done}$$

$$WUR = \ (\text{tx\_lps} = \text{FALSE}) \ \& \ (\text{loc\_wake\_req} = \text{TRUE}) \ \& \ (\text{link\_status} = \text{OK})$$

An LPS and WUR command is transmitted through transmitting a single OAM frame with the respective bit set. Transmission of LPS and WUR does not utilize the OAM handshaking signals (Valid, Toggle, Ack, TogAck) and thus reception of the command is not acknowledged through the OAM handshaking mechanism[6].

Bits D5 and D4 of Symbol 0 must remain reserved (value always 0) at MDIO level. Register reads and writes as part of the 1000BASE-T1 OAM message transmit and receive registers with respective bits set must have no effect on WUR/LPS.

8.2 Wake-Up Pulse (WUP)

The Wake-up pulses (WUP) is a command to indicate a wake-up request to the link partner. The wake-up pulse is transmitted if tx_mode is set to SEND_WUP as indicated in modified PHY Control State Diagram.

The pattern which is transmitted shall be identical to the side stream scrambler used in nominal IEEE 802.3bp tx_mode = SEND_T with the scrambler configured to operate in master mode. The symbol rate shall be 62.5

[6] Acknowledgement happens on wakeup-sleep handshake-level by responding with a LPS command.

MBaud. The differential output voltage over an 100 ohms differential load shall be 1.15V +/- 0.15V peak to peak (with maximum limited per IEEE 97.5.3.5).

The duration of wake-up pulse duration is 1ms +/- 0.3ms to allow reliable detection. The energy detection of a WUP command is left to the implementer.

PHYs with multi-speed capabilities shall use the specified WUP pattern corresponding to the speed the PHY is configured to operate in. The speed configuration process depends on the application and can be through means of pin-strapping, autonegotiation result, register configuration, OTP fuses or similar.

If WUP is sent prior to autonegotiation results are available, then WUP should be the minimum speed advertised by the autonegotiation.

Note, it is only guaranteed that a WUP can be detected reliably if the responder PHY devices supports and operates in the WUP associated speed mode.[7]

[7] For example, a WUP transmitted by a PHY operating in 100BASE-T1 mode is not guaranteed to be detected by a 1000BASE-T1 device and vise-versa.

9 Modified PMA and PCS IEEE802.3bp

This section describes the modification of the Physical Media Attachment and Physical Coding Sublayer of IEEE802.3bp. These modification are to make the Sleep/Wake-up specification be applicable for 1000Base-T1.

9.1 PHY power modes

The following state diagram shows the power state machine which implements the two-way handshake protocol.

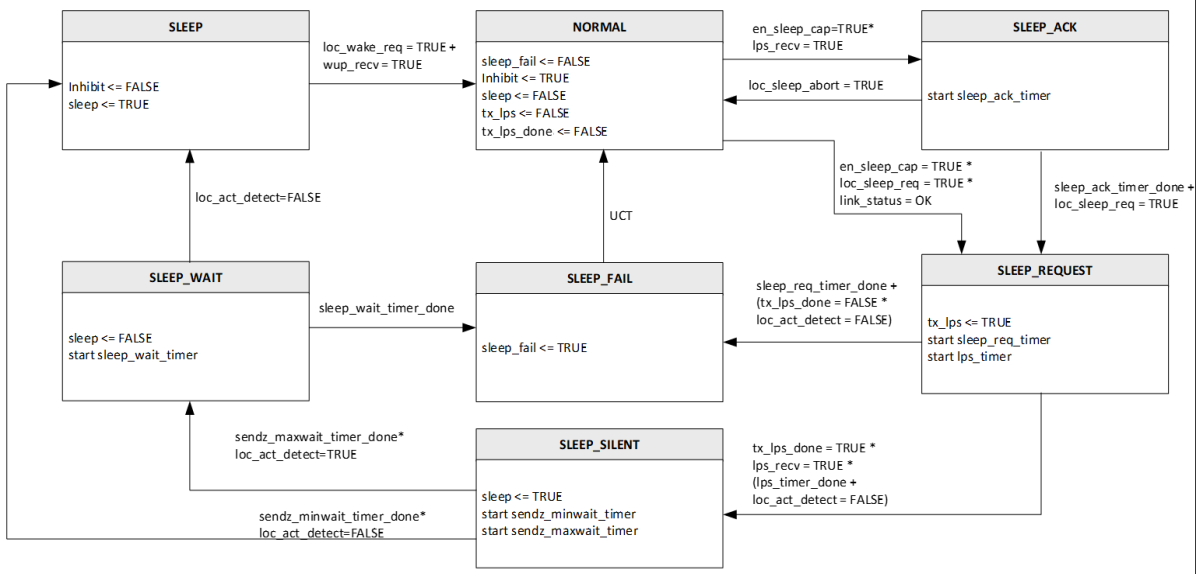


Figure 3 PHY power mode state diagram

9.1.1 PHY reset and initialization

After a device reset the PHY may automatically assert loc_wake_req. This triggers a WUP transmission before normal training is performed.

9.1.2 Sleep

In case the link is up and *Sleep.request* is asserted the PHY will enter the Sleep Request state and will send LPS commands. The link partner receiving those LPS commands enters SLEEP_ACK state and starts sleep_ack_timer. If loc_sleep_abort is asserted, the sleep is aborted because of incoming data message. If sleep reject is not done, the link partner will enter SLEEP_REQUEST state and send LPS commands. If the PHY detects that it has sent and received LPS commands it transits to SLEEP_SILENT state and eventually to SLEEP. On the other hand, if the handshaking is not done before sleep_req_timer timeout, the PHY enters SLEEP_FAIL and back to SEND_DATA state.

9.1.3 Wakeup

The signalling of a *Wakeup.request* depends on the state of the link. If the link is up (tx_mode = SEND_N) the PHY will transmit a WUR command over the active link. If the link is down (link_control = DISABLE and

sync_link_control = DISABLE) the PHY may transmit a WUP pulse. If the link is not yet established (!loc_rcvr_status) for instance because the link is still in training (tx_mode = SEND_T) then a WUR command is sent once the link is established.

The *Wakeup.indicate* shall be generated upon wakeup events. This service primitive is generated in any of the following cases:

- The device is in sleep state and a WUP pulse (wup_recv) is received over MDI
- The link is up and a WUR is received (wur_recv) was signalled
- A local wakeup (loc_wake_req) is asserted

The implementation of the energy detection process that asserts wup_recv is left to the PHY vendor. The energy detection process must not take longer than 2ms. It must be ensured that a transmitted WUP pattern on the link reliably triggers the energy detection (wup_recv=TRUE).

9.1.4 Wakeup-forwarding

Multi-PHY devices (e.g. switches) or PHYs that implement WAKE_FWRD or WAKE_IN_OUT pins shall have a selective wakeup forwarding mechanism. If a multi-PHY device detects a *Wakeup.Request in the form of WUP/WUR*, it must be possible to forward the *indication* to one or multiple other PHYs of the device.

It shall be possible to forward a wakeup from the originating PHY to selectable target PHYs. On these target PHYs the wakeup is send over MDI (as WUP or WUR, depending on the link status)[8].

In case the device implements a WAKE_FWRD or WAKE_IN_OUT pin, a wakeup forwarding shall be indicated by asserting the pin.

9.1.5 Variables

loc_act_detect : Local activity detection signal. The variable is set to FALSE if consecutive symbols of zeros were received; otherwise set to TRUE. The value of loc_act_detect shall be set to TRUE (FALSE) within 1 us.

lps_recv : Set if a LPS command was entirely received.

wur_recv : Set if a WUR command was entirely received.

wup_recv : Set if WUP pulses were sensed.

tx_lps : If set, LPS bits are transmitted.

tx_lps_done : Set after entire OAM frame (12 OAM symbol) containing the LPS command has been transmitted

lps_timer: Timer used to ensure reliable transmission and decoding of LPS command. The timer shall expire 94.504us +/- 0.936us (approximately +/- 1%)

lps_timer_done: Set after lps_timer has expired

loc_sleep_req : Set if a sleep is requested by the local PHY.

loc_wake_req : Set if a wakeup is requested by the local PHY.

[8] In case wakeup events arrive on multiple sources (e.g. pin and MDI) in a short interval, the wakeup event may be joint into a single event.

sleep_req_timer_done: Set when the sleep_req_timer has expired.

sleep_ack_timer_done: Set when the sleep_ack_timer has expired.

loc_sleep_abort: Set if a remote sleep request is to be rejected while still in SLEEP_ACK phase

sleep_fail : Set if a sleep handshake was aborted by the link partner.

inhibit : Set if the (external) power supply shutdown is inhibited.

sleep : Set by the power state machine to notify PHY CTRL to disable transmission.

en_sleep_cap : Indicates whether sleep capability is enabled.

sleep_wait_timer_done: Set when sleep_wait_timer has expired.

9.1.6 Timers

wup_timer : A timer used to wait for reliable detection of WUP pulse. The timer shall expire 1ms +/- 0.3ms after being started.

sleep_ack_timer : A timer used in SLEEP_ACK state to check whether NM decides to reject sleep flow on an incoming data message or not. The timer shall expires 8ms after being started.

sleep_req_timer : A timer set up in SLEEP_REQ to check if the handshaking is properly done by both PHYs. If the PHY doesn't enter SLEEP state before timeout, it enters SLEEP_FAIL state and back to NORMAL. The timer shall expires 16ms after being started.

sendz_minwait_timer: A timer to guarantee a minimum time SEND_Z is transmitted. The timer shall expire after 440 ns (+/- 80 ns).

sendz_maxwait_timer: A timer to limit the maximum number of SEND_Z transmissions to not trip the link partner's PMA watchdog. This timer shall expire after 1.6 us (+/- 120ns).

sleep_wait_timer: A timer to limit the time to stay in SLEEP_WAIT state before going to SLEEP_FAIL state, if local_act_detect is always true during SLEEP_WAIT state. The timer shall expire after 4 us (+/- 120 ns).

9.2 PMA PHY Link Synchronization

The diagram below shows the modified link sync state machine.

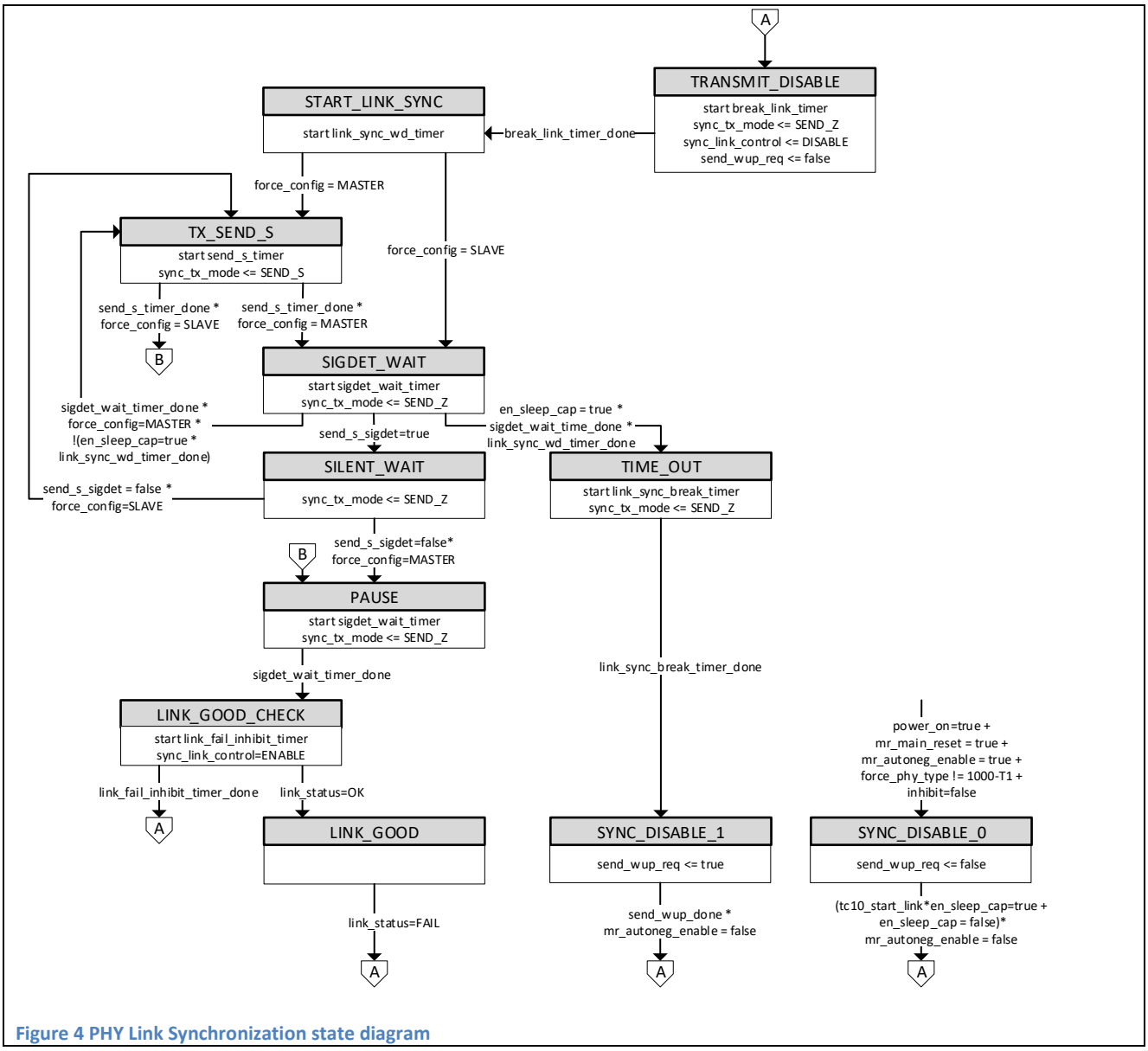


Figure 4 PHY Link Synchronization state diagram

9.2.1 Timers

link_sync_wd_timer: Link sync watchdog timer of 40 ms (+/- 400 us). When link_sync_wd_timer has expired, WUP pattern is transmitted in another attempt to wake up the link partner.

link_sync_break_timer: Link sync break timer of 30 us (+/- 300ns).

9.2.2 Variables

send_wup_req: The variable is set by Link synchronization state machine to request a retransmission of WUP pattern from the PHY control state machine.

tc10_start_link: Asserted by the PHY when it is ready to start link_sync or AutoNeg. The implementation is vendor specific.

9.3 PMA PHY Control State Diagram

The following Figure shows the Modified PMA PHY Control State machine which implements parts of the power sequencing state machine.

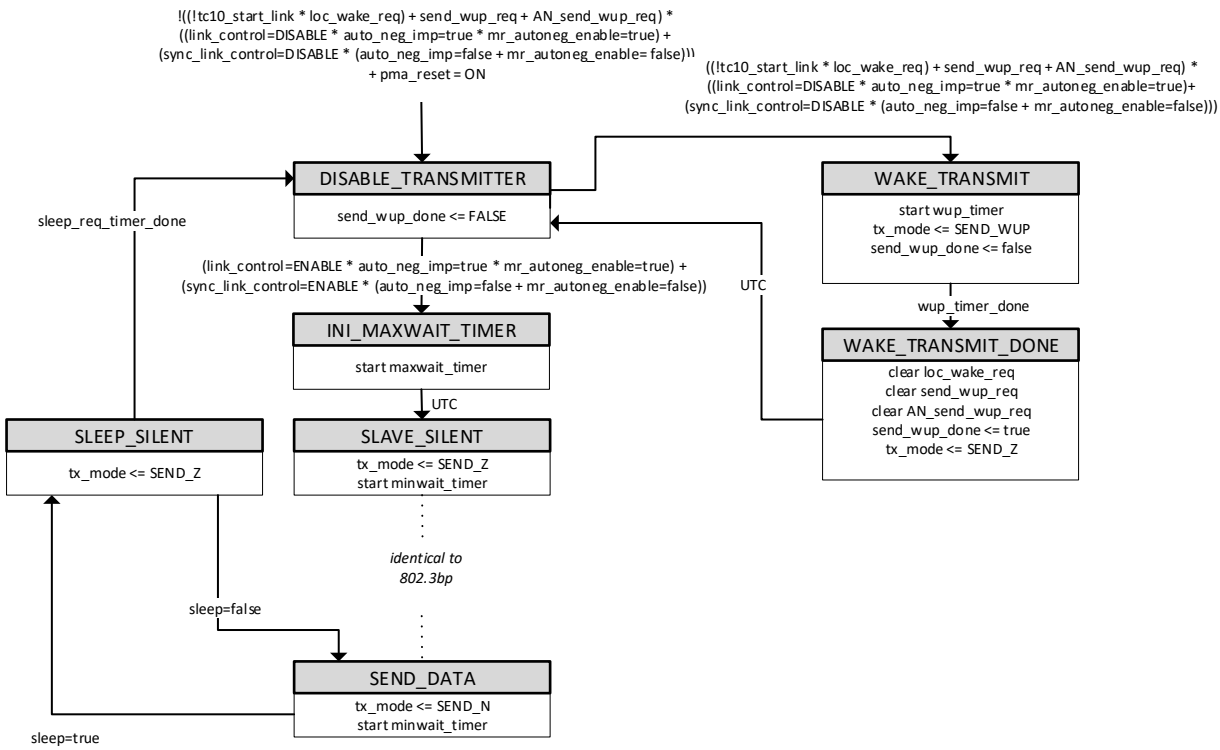


Figure 5 PMA PHY Control state diagram

9.3.1 Variables

sleep : Set by the power statemachine to notify PHY CTRL to disable transmission.

AN_send_wup_req: Set by the AutoNeg arbitration state machine to request a retransmission of a WUP command.

send_wup_req: Set by the PHY Link Synchronization state machine to request a retransmission of a WUP command.

mr_autoneg_enable : controls the enabling and disabling of the Auto-Negotiation function.

auto_neg_imp : This variable indicates if an optional Auto-Negotiation sublayer is associated with the PMA.

link_control : Used by Auto-Negotiation to disable or enable PMA processing.

sync_link_control : Used by Link Synchronization to indicates the data source for the PMA transmit function.

9.3.2 Timers

sleep_req_timer : A timer set up in SLEEP_REQ to check if the handshaking is properly done by both PHYs. If the PHY doesn't enter SLEEP state before timeout, it enters SLEEP_FAIL state and back to NORMAL. The timer shall expires 16ms after being started.

wup_timer : This timer controls the duration for which WUP is transmitted. The timer shall expire 1ms +/- 0.3ms after being started.

9.4 Auto Negotiation state machine

The following Figure shows and Auto Negotiation state diagram with modifications for the sleep/wake up applications.

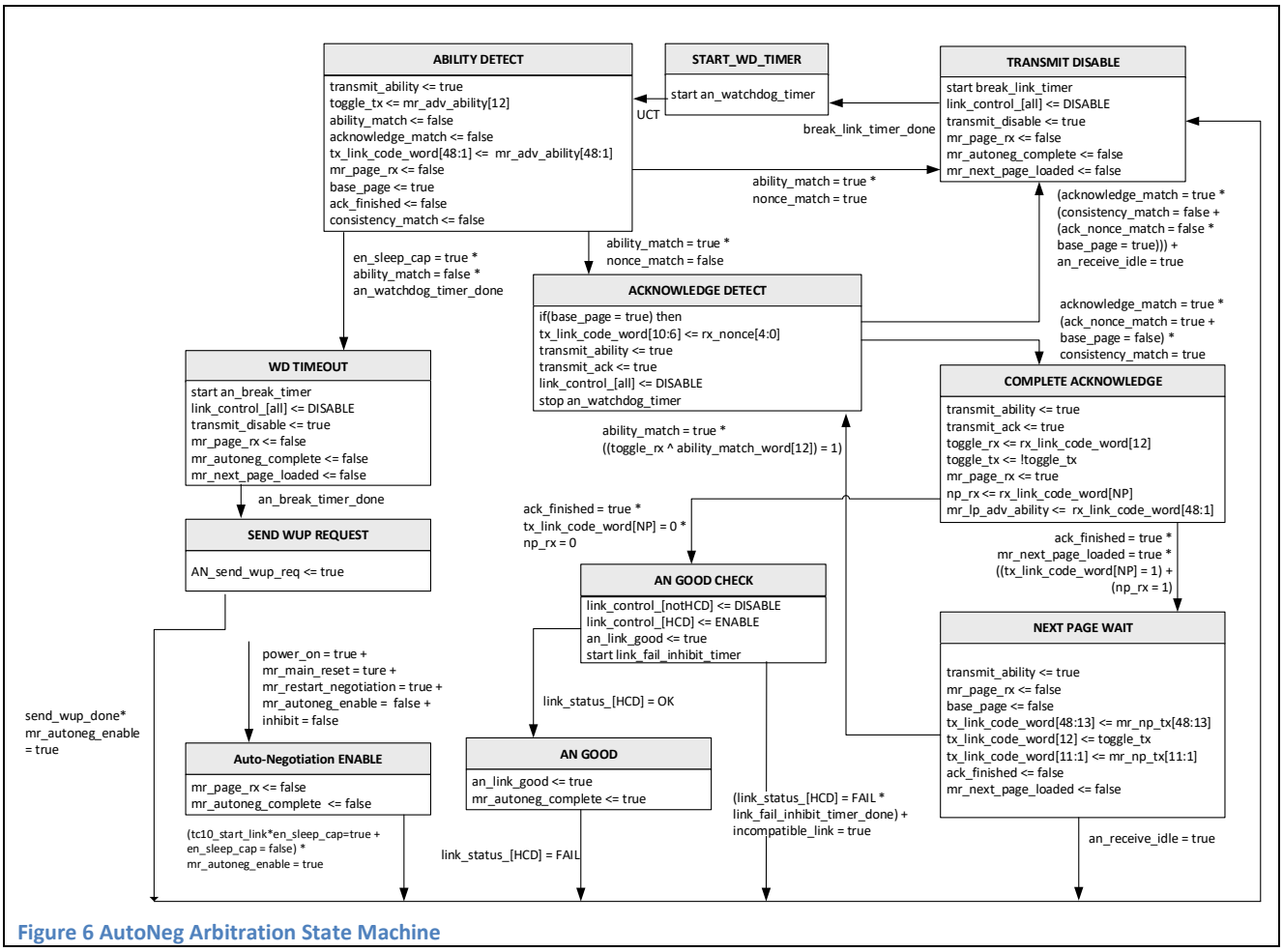


Figure 6 AutoNeg Arbitration State Machine

9.4.1 Variables

AN_send_wup_req: Set by the AutoNeg arbitration state machine to request a retransmission of a WUP command.

loc_wake_req : Set if a wakeup is requested by the local PHY.

sleep : Set by the power statemachine to notify Auto Negotiation and Link Synchronization to disable transmission.

tc10_start_link: Asserted by the PHY when it is ready to start link_sync or AutoNeg. The implementation is vendor specific.

9.4.2 Timers

an_watchdog_timer: Autoneg watchdog timer to trigger WUP retransmission. Timer shall be 40 ms (+/- 400 us).

AN_break_timer: Autoneg break timer shall be 30 us (+/- 300 ns).